

WHAT IS CLAIMED IS:

1. A method of providing fee-based access to data, comprising:
providing an abstract model for logically defining abstract operations to access the data, the abstract model comprising:
 - (i) a plurality of logical fields;
 - (ii) a mapping rule for each of the plurality of logical fields, which map the plurality of logical fields to physical entities of the data; and
 - (iii) a fee schedule for each of the plurality of logical fields, wherein each fee schedule for a given logical field defines a fee to be charged when the given logical field is involved in an abstract operation to access a physical entity corresponding to the given logical field.
2. The method of claim 1, further comprising:
accessing the data according to an abstract operation comprising at least two of the plurality of logical fields; and
calculating the fee to be charged based on separate fee schedules corresponding to each of the at least two plurality of logical fields.
3. The method of claim 1, further comprising:
providing a run-time component configured with transformation instructions to transform each abstract operation, comprising logical fields selected from the plurality of logical fields, into a physical operation consistent with the physical data; and
providing a fee calculator configured to calculate a fee for executing physical operations based on the fee schedules.
4. The method of claim 1, wherein the abstract operation is an abstract query comprising at least two logical fields, the method further comprising:

accessing a corresponding fee schedule for each of the at least two logical fields;

determining a per request fee for a first one of the at least two logical fields, wherein the per request fee is charged for each abstract operation involving the first one of the at least two logical fields; and

determining a per item fee for a second one of the at least two logical fields, wherein the per item fee is charged for each instance of the second one of the at least two logical fields involved in a given abstract operation.

5. The method of claim 4, further comprising multiplying the per item fee by a number of instances of the second one of the at least two logical fields to determine a product, and summing the product and the per request fee to determine the fee to be charged.

6. The method of claim 1, wherein at least one fee schedule defined by the abstract model specifies a first fee for a first type of operation and a second fee for a second type of operation; and further comprising calculating the fee to be charged based on the type of operation performed.

7. The method of claim 6, wherein the first type of operation is a query and the second type of operation is one of an insert and an update.

8. A method of providing fee-based access to physical data comprising a plurality of physical entities each comprising a plurality of physical fields, the method comprising:

providing an abstract model for defining abstract operation specifications logically describing operations to access the data, the abstract model comprising:

(a) a plurality of logical fields;

- (b) a mapping rule for each of the plurality of logical fields, which map each of the plurality of logical fields to at least one of the physical entities of the data;
 - (c) a plurality of model entity definitions, each comprising at least one logical field corresponding to a physical field of a physical entity; and
 - (d) a logical field fee schedule for each of the plurality of logical fields, wherein the fee schedules each specify a fee for accessing a corresponding physical field as part of a physical operation.
9. The method of claim 8, further comprising:
- transforming, according to the abstract model, abstract operation specifications into physical operation specifications consistent with the physical data, wherein each abstract operation specification includes at least one of the plurality of model entity definitions; and
 - calculating a fee for executing physical operations based on the fee schedules.
10. The method of claim 8, wherein at least one logical field fee schedule defined by the abstract model specifies a first fee for a first type of operation and a second fee for a second type of operation.
11. The method of claim 10, wherein the first type of operation is a query and the second type of operation is one of an insert and an update.
12. A method of providing fee-based access to data comprising a plurality of physical entities, each comprising a plurality of physical fields, comprising:
- receiving instructions to perform an operation for accessing the data;
 - performing the operation;
 - determining field-specific fees for each of a plurality of the physical fields accessed by the operation; and
 - calculating a total fee to be charged to a user for the operation.

13. The method of claim 12, wherein the physical entities are database tables.

14. The method of claim 12, wherein the operation is one of a query, an insert and an update.

15. The method of claim 12, wherein determining field-specific fees comprises determining whether a field-specific fee is a per request fee or a per item fee, wherein the per request fee is a singular fee charged for the operation regardless of a number of instances a corresponding physical field is included in the operation and wherein the per item fee is charged for each instance of a corresponding physical field included in the operation.

16. The method of claim 12, wherein determining field-specific fees comprises accessing fee schedules for each respective physical field accessed by the operation.

17. The method of claim 16, wherein each of the fee schedules defines a separate fee for each separate operation type.

18. The method of claim 17, wherein the separate operation types comprise queries, inserts and updates.

19. The method of claim 12, wherein determining field-specific fees comprises accessing an abstract model for logically defining the operation accessing the data, the abstract model comprising:

- (i) a plurality of logical fields;
- (ii) a mapping rule for each of the plurality of logical fields, which map the plurality of logical fields to physical entities of the data; and
- (iii) a fee schedule for each of the plurality of logical fields, wherein each fee schedule for a given logical field defines a fee to be charged

when the given logical field is involved in an abstract operation to access a physical entity corresponding to the given logical field.

20. The method of claim 19, wherein each mapping rule comprises an access method for each logical field of an abstract operation specification logically defining the operation accessing the data, the access method describing a physical location of a physical entity.

21. The method of claim 19, wherein each fee schedule defines at least one of a per request fee and a per item fee, wherein the per request fee is a singular fee charged for the operation regardless of a number of instances a corresponding logical field is included in the operation and wherein the per item fee is charged for each instance of a corresponding logical field included in the operation.

22. The method of claim 19, further comprising:
transforming, according to the abstract model, abstract operations into physical operation consistent with the physical data, wherein each abstract operation includes at least one of the plurality of model entity definitions; and
calculating the fee for executing physical operations based on the fee schedules.

23. The method of claim 19, further comprising:
accessing the data according to the abstract operation, the abstract operation comprising at least two of the plurality of logical fields; and
calculating the fee to be charged based on separate fee schedules corresponding to each of the at least two plurality of logical fields.

24. The method of claim 19, further comprising:
providing a run-time component configured with transformation instructions to transform each abstract operation, comprising logical fields selected from the plurality of logical fields, into a physical operation consistent with the physical data; and

providing a fee calculator configured to perform the calculating of the fee for executing physical operations based on the fee schedules.

25. The method of claim 19, wherein the abstract operation is an abstract query comprising at least two logical fields, the method further comprising:

accessing a corresponding fee schedule for each of the at least two logical fields;

determining a per request fee for a first one of the at least two logical fields, wherein the per request fee is charged for each abstract operation involving the first one of the at least two logical fields; and

determining a per item fee for a second one of the at least two logical fields, wherein the per item fee is charged for each instance of the second one of the at least two logical fields involved in a given abstract operation.

26. The method of claim 25, further comprising multiplying the per item fee by a number of instances of the second one of the at least two logical fields to determine a product, and summing the product and the per request fee to determine the fee to be charged.

27. The method of claim 19, wherein at least one fee schedule defined by the abstract model specifies a first fee for a first type of operation and a second fee for a second type of operation; and further comprising calculating the fee to be charged based on the type of operation performed.

28. The method of claim 27, wherein the first type of operation is a query and the second type of operation is one of an insert and an update.

29. A computer-readable medium containing a program which, when executed by a processor, performs operations for accessing physical data comprising a plurality of physical entities, each having a plurality of physical fields, the operation comprising:

receiving instructions to perform an operation accessing the data;
causing performance of the operation;
determining field-specific fees for each of a plurality of the physical fields
accessed by the operation; and
calculating a total fee to be charged to a user for the operation.

30. The computer-readable medium of claim 29, wherein the physical entities are database tables.

31. The computer-readable medium of claim 29, wherein the operation is one of a query, an insert and an update.

32. The computer-readable medium of claim 29, wherein determining field-specific fees comprises determining whether a field-specific fee is a per request fee or a per item fee, wherein the per request fee is a singular fee charged for the operation regardless of a number of instances of a corresponding physical field are included in the operation and wherein the per item fee is charged for each instance of a corresponding physical fee included in the operation.

33. The computer-readable medium of claim 29, wherein determining field-specific fees comprises accessing fee schedules for each respective physical field accessed by the operation.

34. The computer-readable medium of claim 33, wherein each of the fee schedules defines a separate fee for each separate operation type.

35. The computer-readable medium of claim 34, wherein the separate operation types comprise queries, inserts and updates.

36. The computer-readable medium of claim 29, wherein determining field-specific fees comprises accessing an abstract model for logically defining the operation accessing the data, the abstract model comprising:

- (i) a plurality of logical fields;
- (ii) a mapping rule for each of the plurality of logical fields, which map the plurality of logical fields to physical entities of the data; and
- (iii) a fee schedule for each of the plurality of logical fields, wherein each fee schedule for a given logical field defines a fee to be charged when the given logical field is involved in an abstract operation to access a physical entity corresponding to the given logical field.

37. The computer-readable medium of claim 36, wherein each mapping rule comprises an access method for each logical field of an abstract operation specification logically defining the operation accessing the data, the access method describing a physical location of a physical entity.

38. A method for constructing abstract queries defined by a plurality of logical fields which map to a plurality of physical entities of physical data having a particular physical data representation in a database, the method comprising:

receiving user input via a user interface, the input comprising a reference to a model entity definition comprising: (i) two or more logical fields each corresponding to a separate physical entity; and (ii) a fee schedule for accessing physical entities based on the model entity definition;

based on the model entity definition, selectively adding at least one of the two or more logical fields to an abstract query;

receiving a plurality of abstract query contributions for the abstract query, wherein the plurality of abstract query contributions are defined by selected logical fields and a corresponding value for each of the selected logical fields;

receiving a plurality of result fields for the abstract query, wherein the plurality of result fields is defined by selected logical fields;

converting the abstract query into a physical query consistent with the particular physical data representation of the data;
executing the physical query; and
calculating, on the basis of the fee schedule, a fee to charge for execution of the physical query.

39. The method of claim 38, wherein each of the plurality of physical entities is a table in a database.

40. The method of claim 38, wherein selectively adding the at least one of the two or more logical fields comprises:

determining whether the at least one logical field is a required field as specified by the model entity definition; and
if so, adding the at least one logical field to the abstract query.

41. The method of claim 38, wherein a per request fee is defined for a first one of the two or more logical fields and a per result fee is defined for a second one of the two or more logical fields and wherein calculating the fee comprises: (i) calculating a product of the per result fee and a number of results for the second one of the two or more logical fields; and (ii) summing the product and the per request fee.

42. The method of claim 38, wherein converting the abstract query into the physical query comprises mapping each of the logical fields of the abstract query to respective physical entities of the physical data.

43. A method for modifying physical data comprising a plurality of physical entities and having a particular physical data representation in a database, the method comprising:

receiving a selection of an abstract modification operation;

receiving a selection of a model entity definition on which to perform the abstract modification operation, the model entity definition comprising two or more logical fields each corresponding to a separate physical entity;

based on at least the received selections, generating at least two physical modification statements, each modifying one of the two separate physical entities of the physical data;

ordering the at least two physical modification statements;

executing modification operations according to the physical modification statements, whereby the data is modified; and

calculating a fee to charge for executing the modification operations based on a defined fee schedule for the model entity definition.

44. The method of claim 38, wherein the ordering is done according to a physical entity relationships specification defining hierarchical relationships between physical entities of the data to ensure integrity of the data.

45. The method of claim 38, wherein building the at least two physical modification operations is done according to mapping rules which map the two or more logical fields to their corresponding physical entity.

46. A method of providing a logical framework for defining abstract operations for accessing physical data comprising a plurality of physical entities each comprising a plurality of physical fields, the method comprising:

providing an abstract model for defining abstract operation specifications logically describing operations to access the data, the abstract model comprising:

(a) a plurality of logical fields;

(b) a mapping rule for each of the plurality of logical fields, which map each of the plurality of logical fields to at least one of the physical entities of the data;

(c) a plurality of model entity definitions, each comprising at least one logical field corresponding to a physical field of a physical entity; and

(d) model entity fee schedules for each of the plurality of model entity definitions, wherein the fee schedules each specify a fee for accessing a physical field of the corresponding model entity definition; and

providing a run-time component to transform, according to the abstract model, abstract operation specifications into physical operation specifications consistent with the physical data, wherein each abstract operation specification includes at least one user-selected model entity definitions of the plurality of model entity definitions.

47. The method of claim 46, wherein each of the plurality of physical entities is a table in a database.

48. The method of claim 46, wherein each physical operation specification is selected from one of an insert statement and an update statement and wherein the model entity fee schedules define different fees for each statement.

49. The method of claim 46, wherein each physical operation specification is a query, and wherein the model entity fee schedules define fees specific to queries.

50. The method of claim 46, wherein the abstract operation specification is an abstract query, and further comprising:

receiving, via a user interface, the abstract query comprising a plurality of query conditions, result fields and a selection of one of the model entity definitions;

accessing the model entity definition corresponding to the selection;

determining whether the model entity definition corresponding to the selection specifies one or more required result fields; and

if so, adding the one or more required result fields to the query.

51. The method of claim 46, further comprising transforming, by the run-time component transforms and according to the abstract model, a single abstract operation specification into at least two separate physical operation specifications consistent with the physical data, wherein each physical operation specification modifies a different physical entity of the data and wherein each physical operation specifications is ordered for execution according to a physical entity relationships specification defining hierarchical relationships between the physical entities of the data.

52. The method of claim 46, further comprising:
issuing, by a requesting entity, a request to execute a single abstract operation specification; and
transforming, by the run-time component, the single abstract operation specification into the at least two physical operation specifications for modifying the data.

53. The method of claim 52, wherein transforming the single abstract operation specification into the at least two physical operation specifications comprises:
generating the at least two physical operation specifications; and
ordering the at least two physical operation specifications according to a physical entity relationships specification of the abstract model.

54. The method of claim 46, wherein each mapping rule comprises an access method for one of the plurality of logical fields.

55. The method of claim 54, wherein the access method describes a location of the physical entities of the data.

56. The method of claim 46, further comprising a logical field fee schedule for each of the plurality of logical fields, wherein the fee schedules each specify a fee for accessing a corresponding physical field as part of a physical operation specification.

57. The method of claim 56, wherein at least one fee schedule defined by the abstract model specifies a first fee for a first type of operation and a second fee for a second type of operation.

58. The method of claim 57, wherein the first type of operation is a query and the second type of operation is one of an insert and an update.

59. A computer-readable medium containing a program which, when executed by a processor, provides a logical framework for defining abstract query operations, the program comprising:

an abstract model for defining abstract queries logically describing operations to query the data, the abstract model comprising:

- (i) a plurality of logical fields;
- (iv) a mapping rule for each of the plurality of logical fields, which map the plurality of logical fields to physical entities of the data; and
- (v) a fee schedule for each of the plurality of logical fields;

a run-time component configured with transformation instructions to transform an abstract query, comprising logical fields selected from the plurality of logical fields, into a physical query consistent with the physical data; and

a fee calculator configured to calculate a fee for executing physical queries based on the fee schedules.

60. The computer-readable medium of claim 59, wherein the mapping rules comprise an access method for each of the plurality of logical fields, wherein the access method describes a location of the physical entities of the data.

61. A computer comprising a memory and at least one processor, and further comprising a logical framework for defining abstract modification operations for modifying physical data, the logical framework comprising:

an abstract model for defining an abstract modification specification logically describing an operation to modify the data, the abstract model comprising:

- (i) a plurality of logical fields;
- (ii) a mapping rule for each of the plurality of logical fields, which map the plurality of logical fields to physical entities of the data; and
- (iii) a fee schedule for each of the plurality of logical fields;

a run-time component to transform an abstract query, comprising logical fields selected from the plurality of logical fields, into a physical query consistent with the physical data; and

a fee calculator configured to calculate a fee for executing physical queries based on the fee schedules.

62. The computer of claim 61, wherein each fee schedule defines at least one of a per request fee and a per item fee, wherein the per request fee is a singular fee charged for the operation regardless of a number of instances a corresponding logical field is included in the operation and wherein the per item fee is charged for each instance of a corresponding logical field included in the operation.

63. The computer of claim 61, wherein the mapping rules comprise an access method for each of the plurality of logical fields, wherein the access method describes a location of the physical entities of the data.

64. A method of providing fee-based access to data comprising a plurality of physical entities, each comprising a plurality of physical fields, comprising:

receiving, via a user interface, user input comprising instructions for an operation for accessing the data selected fields of the plurality of the physical fields;
determining field-specific fees for each of the selected fields;
calculating a fee to be charged to a user for accessing the selected fields; and
displaying the fee to the user via a user interface.

65. The method of claim 64, wherein the physical entities are database tables.

66. The method of claim 64, wherein determining field-specific fees comprises determining whether a field-specific fee is a per request fee or a per item fee, wherein the per request fee is a singular fee charged for the operation regardless of a number of instances a corresponding physical field is included in the operation and wherein the per item fee is charged for each instance of a corresponding physical field included in the operation.

67. The method of claim 64, wherein determining field-specific fees comprises accessing fee schedules for each respective physical field accessed by the operation.

68. The method of claim 67, wherein each of the fee schedules defines a separate fee for each separate operation type.

69. The method of claim 68, wherein the separate operation types comprise queries, inserts and updates.

70. A method for displaying fee information for fee-based access to data comprising a plurality of physical entities, each comprising a plurality of physical fields, comprising:
displaying one or more user interface screens for construction of queries;
receiving, via the one or more user interface screens, user input defining a query configured to access selected fields of the plurality of physical fields; and
displaying, via the one or more user interface screens, a field-specific access fee for each of the selected fields.

71. The method of claim 70, further comprising displaying a per query fee, the per query fee being a singular fee charged for the query regardless of a number of instances a corresponding physical field is included in the query.

72. The method of claim 70, further comprising determining the field-specific fees by accessing an abstract model for logically defining the query, the abstract model comprising:

- (i) a plurality of logical fields;
- (ii) a mapping rule for each of the plurality of logical fields, which map the plurality of logical fields to physical entities of the data; and
- (iii) a fee schedule for each of the plurality of logical fields, wherein each fee schedule for a given logical field defines a fee to be charged when the given logical field is involved in a query to access a physical entity corresponding to the given logical field.

73. The method of claim 72, wherein each mapping rule comprises an access method for each logical field of an abstract operation specification logically defining the operation accessing the data, the access method describing a physical location of a physical entity.

74. The method of claim 72, wherein each fee schedule defines at least one of a per request fee and a per item fee, wherein the per request fee is a singular fee charged for the query regardless of a number of instances a corresponding logical field is included in the query and wherein the per item fee is charged for each instance of a corresponding logical field included in the query.